NISTIR 5176

Optimization of Adaptive Resonance Theory Network With Boltzmann Machine

Omid M. Omidvar
Charles L. Wilson

U.S. DEPARTMENT OF COMMERCE
Technology Administration
National Institute of Standards
and Technology
Computer Systems Laboratory
Advanced Systems Division
Gaithersburg, MD 20899

NIST

NISTIR 5176

Optimization of Adaptive Resonance Theory Network With Boltzmann Machine

Omid M. Omidvar
Charles L. Wilson

U.S. DEPARTMENT OF COMMERCE
Technology Administration
National Institute of Standards
and Technology
Computer Systems Laboratory
Advanced Systems Division
Gaithersburg, MD 20899

April 1993

U.S. DEPARTMENT OF COMMERCE
Ronald H. Brown, Secretary

NATIONAL INSTITUTE OF STANDARDS
AND TECHNOLOGY
Raymond G. Kammer, Acting Director

OPTIMIZATION OF ADAPTIVE RESONANCE THEORY NETWORK WITH BOLTZMANN MACHINE

Omid M. Omidvar,Computer Science Dept.
University of the District of Columbia
Washington, DC 20008
C L. Wilson, National Institute of Standards and Technology
Gaithersburg, MD 20899

Abstract

Optimization of large neural networks is essential in improving the network speed and generalization power, while at the same time reducing the training error and the network complexity. Boltzmann methods have been used as a statistical method for combinatorial optimization and for the design of learning algorithms. In the networks studied here, the Adaptive Resonance Theory (ART) serves as a connection creation operator and the Boltzmann method serves as a competitive connection annihilation operator. By combining these two methods it is possible to generate small networks that have similar testing and training accuracy and good generalization from small training sets. Our findings demonstrate that for a character recognition problem the number of weights in a fully connected network can be reduced by over 80%. We have applied the Boltzmann criteria to differential pruning of the connections which is based on the weight contents rather than on the number of connections.

1 Introduction

Most optimization strategies are a trade-off between error and network complexity. Many known optimization schemes [1, 2, 3, 4] have used this trade-off to minimize the cost function. Boltzmann methods have been used as a statistical method for combinatorial optimization and for the design of learning algorithms [5, 6]. These methoda have also been used in conjunction with a supervised learning method to dynamically reduce network size [7]. The strategy used in this research is to create a network using ART [8] and to remove the weights using Boltzmann criteria during the training process. The ART algorithm was originally implemented in a parallel environment and it was applied to character recognition [9].

This implementation of the ART algorithm consists of finding two sets of weights. Each active memory location is used to store bottom-up and top-down weights which have an optimal resonance with one of the input images. At the beginning of the learning process all memory locations are blank. As learning progresses, the two sets of weights in each active memory location are updated in parallel for all active images. When an image meeting the vigilance constraint is found, the appropriate weights are updated. The vigilance parameter provides a correlation measure of the association strength between the image and the learned weights. If no resonance is achieved for a given image, a new memory location is added to the active list and the weights of this blank location are updated using the image. This process is continued with each image until all available memory location are set and or all of the input images are used.

The Boltzmann method seeks to minimize the number of weights while maintaining the information content of the network. The ART method seeks to minimize an error function on the training set. The important controlling parameter for the Boltzmann method is the information in the network and the iteration time, t, as t approaches infinity. The controlling informational parameter for the ART method is the information provided at $t = 0$ in the initial weights. The algorithmic control in the Boltzmann method is the temperature sequence applied during the iteration. The equivalent controlling parameter for the ART is the vigilance.

The ART network is used as a starting network for pruning. The pruning is carried out by selecting a normalized temperature, and discarding the weights is based on a removal probability calculated via the Boltzmann method. The probability of the removal is compared to a set of uniformly distributed random numbers. If the calculated probability is greater than the random numbers then the corresponding weight is set to zero. The process is carried out for each iteration of ART. If a weight is removed it may subsequently be restored by the ART algorithm; the restored weight may survive if it has sufficient magnitude in subsequent iterations. The result of the temperature changes indicates that a network of reduced size can perform as good as, or in some cases better than, the fully converged initial network. The effect of changes in the vigilance parameter on the pruning and restoration process is studied. The optimized network, which is considerably smaller in size, has a higher speed for testing and training, and yields better generalization than the unpruned ART network.

2 ART Architecture

2.1 ART-1 Learning

The ART-1 algorithm developed by Carpenter and Grossberg [8] is ideally suited for self-organization of unconstrained fonts or hand printed characters. The calculations involved are well adapted to parallel implementation on a single bit processor and are naturally parallel across the image field. The specific implementation of ART-1 for the parallel array processor is shown in figure 1. All of the three-line paths involve operations on 1024 bits of image data in parallel and represents parallel transfer of data.

2.1.1 Parallel Weight Selection

The ART-1 algorithm finds two sets of weights, $\mathbf{Z}$'s, for each of j active memory locations (each memory location is used to store a pair of bottom-up and top-down weights) which have an optimal resonance with one of the i input images. At the beginning of the learning process all memory locations are blank. As learning progresses, two sets of weights $\mathbf{Z}_{up,j}$ and $\mathbf{Z}_{down,j}$, for the j active memory locations are updated for each of i images, $\mathbf{I}_i$, by calculating:

$$\mu_{i,j} = \mathbf{I}_i \cdot \mathbf{Z}_{up,j} \tag{1}$$

$$\bar{x}_i = \mathbf{I}_i \cdot \mathbf{I}_i \tag{2}$$

$$\bar{T}_{i,j} = \mathbf{I}_i \cdot \mathbf{Z}_{down,j} / \bar{x}_i \tag{3}$$

and finding the maximum μ for which $\bar{T} > \rho$, the vigilance parameter of the required resonance. $\bar{T}$ provides a correlation measure of the association strength between the image, $\mathbf{I}$, and the learned weights $\mathbf{Z}_{down}$. When an image meeting the vigilance parameter constraint is found, the appropriate weights are adapted using:

$$\mathbf{Z}_{up,j} = \mathbf{I}_i \cdot \mathbf{Z}_{down,j} / (0.5 + \bar{T}_{i,j}) \tag{4}$$

and

$$\mathbf{Z}_{down,j} = \mathbf{I}_i \cdot \mathbf{Z}_{down,j} \tag{5}$$

If no resonance is achieved for a given image, a new memory location is added to the active list and the weights of this "blank" location are adapted to the image. This process is continued with each image until all available memory locations are set or all of the input images in the training set are used. After all vacant memory locations are used, each memory location is compared to the product $\mu_{i,j}\bar{T}_{i,j}$ and the memory location for which this product is largest is then updated.

2.1.2 Evaluation of Self-Organization

The evaluation of the self-organized classes is achieved by accumulation of statistics in a classification variable:

$$Z_{class,k,j} = Z_{class,k,j} + 1 \ \text{ if } \ \text{class of}(\mathbf{I}_i) = k \tag{6}.$$

This table can then be used to determine the maximum selection strength of each memory location for all images and to assign classes to images based on resonance performance achieved over all memory location and class assignments. This allows a new set of images to be learned wholly by example and divided into classes based on the recognition results achieved using the images assigned to the training set.

3 Recognition

Recognition is achieved by finding the maximum strength of resonance for the weighted classes, $\max(w_{k,j})$, using:

$$w_{k,j} = \sum_{i=k} Z_{class,k,j} \times \mu_{i,j}\bar{T}_{i,j} \tag{7}.$$

In addition, the average resonance strength of the strongest weighted resonance,

$$P = \max(w_{k,j}/n_{w,k,j}), \tag{8}$$

where $n_{w,k,j}$ is the number of terms used to form each $w_{k,j}$, provides a confidence limit for the evaluation of classification errors. If the value of P is less than the confidence expected for correctly classified data in the training set, then items should be classified as unknown.

This procedure for detecting incorrect classifications is in such a way that both detectable errors, labeled "Unknown", and undetectable errors, labeled "Wrong" are considered. This same process will label some correct classifications as either "Unknown" or "Wrong". This is unavoidable since some correct classifications may have low confidence. Detectable errors, which are found by imposing confidence limits on the association strength, P, are separated from undetectable errors by examining the cumulative distributions of recognition rate with respect to the association strength, for example, all matches exceed an association strength of 0.5 and no match exceeds an association strength of 0.875. If all matches with association strength greater than 0.64 are accepted, all incorrect matches are detected and no correct matches are lost. In this case, no correct classifications are lost in the rejection process.

4 Data Set

The test sample consisted of 300 machine printed digits taken from a single set of laser printer output. The primary source of variation in the test sample can be traced to variation in thresholding during scanning and segmentation. The digits were not centered in the field or scaled to fit the 32 by 32 image size used in feature extraction. Each test set is divided into two 150 character samples. The first 150 characters are used for the construction of the optimized ART-1 weights and the Z_{class} statistical array. The second 150 image samples are then used to test the classification of previously unseen images for maximum resonance. The image representation of the bottom up weights and the digits used in learning these weights for unfiltered machine print characters with and without optimization are shown in figures 2 and 3 respectively. The reason that the character images in figure 2 are much less clear than the images in figure 3 is that the pruning method is independent of the positional associations present in the image data of the training set.

5 Optimization Process

The ART network is used as a starting point for the Boltzmann weight pruning algorithm. The pruning was carried out by selecting a normalized temperature, T, and removing weights based on a probability of removal.

$$P_i = 1 - exp(-|w_i|/T) \tag{9}$$

The values of P_i are compared to a set of uniformly distributed random numbers, R_i, on the interval [0,1]. If the probability P_i is greater than R_i then the weight is set to zero. The process is carried out for each iteration of the ART algorithm during the training process and is dynamic. If a weight is removed it may subsequently be restored by the ART algorithm; the restored weight may survive if it has sufficient magnitude in subsequent iterations.

As the size of the temperature change increases the number of weights removed initially increases, but the effect of later iterations of optimization and pruning is to decrease the rate at which weights are removed. The critical temperature, T_c, is the temperature at which recognition rate of the network is at a maximum. Any increase in T beyond T_c results in a decrease in recognition. The changes in the number of connections with respect to the iteration time t at two temperatures during the optimization process are shown in figures 4 and 6. The temperature of 0.5 shown in figure 4 is less than T_c. The temperature of

0.8 shown in figure 6 is greater than T_c. Below T_c the ART learning recreates most of the connections after each learning cycle. This causes the oscillation shown in figure 4. Above T_c the ART learning process can't keep up with the more rapid pruning and several learning cycles are required for a single cycle of weight creation and destruction. This is shown in figure 6.

The effect of changes in the vigilance parameter and two temperature on the performance of the network is also investigated. As the temperature is increased the accuracy of the network for recognition decreases slowly for temperatures up to 0.4. As the temperature approaches 0.5 the rate of weight removal slows and the rate of accuracy decay accelerates. The curves for $T = 0.5$ and $T = 0.8$ are shown in figures 5 and 7. Each point on these curves is the result of a calculation of the type shown in figures 4 and 6. The critical temperatue is estimated to be 0.58. Testing accuracy below this temperature is not strongly effected by the pruning process. Accuracy, for networks pruned above T_c is reduced. The peak accuracy at $T = 0.5$ is is 91% when $\rho = 0.7$. The peak accuracy at $T = 0.8$ is is 70% when $\rho = 0.6$.

In a given training cycle some weights are removed. If these weights are redundant they will be compensated by other weights in the network. If these weights are critical they will be restored by the ART optimization. At T_c, the ART creation process is just balanced by the Boltzmann pruning. To evaluate the generalization capability of the pruned network, the network associated with a temperature $T = 0.5$ was tested on a sample of 150 digits. The training accuracy was 90%; the accuracy achieved in the test was 91%. This is consistent with good generalization with a value of $T = 0.5$. The optimal vigilance value was around 0.7. The highest recognition rate was achieved with these values.

6 Conclusions

A method of network optimization has been developed which reduces the number of weights required for moderately accurate character recognition by 80%. The method is based on achieving equilibrium between the information in the training set and the number of network weights by concurrent weight creation using ART learning and Boltzmann optimization by weight removal. These reductions allow both smaller training sets and smaller classification networks to be used. This type of optimization was much more sucessful when applied to MLP network [7] because in these networks the ordering of the information in the network is unrelated to the ordering of information in the learnings set. In ART, there is a direct positional association between the image information stored in the network and the training set.

References

[1] E. B. Baum and D. Haussler. What size net gives valid generalization? *Neural Comput*, 1:151–160, 1989.

[2] M. C. Mozer and P. Smolensky. Using relevance to reduce network automatically. *Conn Science*, 1:3–16, 1989.

[3] Y. Le Cun, J. S. Denker, and S. A. Solla. Optimal Brain Damage. In D. Touretzky, editor, *Advances in Neural Information Processing Systems*, volume 2, pages 396–404. Morgan Kaufman, 1990.

[4] I. Guyon, V. N. Vapnick, B. E. Boser, L. Y. Botton, and S. A. Solla. Structural risk minimization for character recognition. In R. Lippmann, editor, *Advances in Neural Information Processing System*, volume 4, pages 471–479. Morgan Kauffman, 1992.

[5] D. H. Ackley, G. E. Hinton, and T. J. Seynowski. A learning algorithm for Botlzmann machines. *Cognitive Science*, 9:147–169, 1985.

[6] S. Kirkpatrick, C. D. Gelatt, and M. P. Vacchi. Optimization by simulated annealing. *Science*, 220:671–680, 1983.

[7] O. M. Omidvar and C. L. Wilson. Optimization of neural network topology and information content using boltzmann methods. In *Proceedings of the IJCNN, volume IV*, pages 594–599, June 1992.

[8] G. A. Carpenter and S. Grossberg. A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics, and Image Processing*, 37:54–115, 1987.

[9] C. L. Wilson, R. A. Wilkinson, and M. D. Garris. Self-organizing neural network character recognition using adaptive filtering and feature extraction. *Progress in Neural Networks*, 3, 1991. to be published.

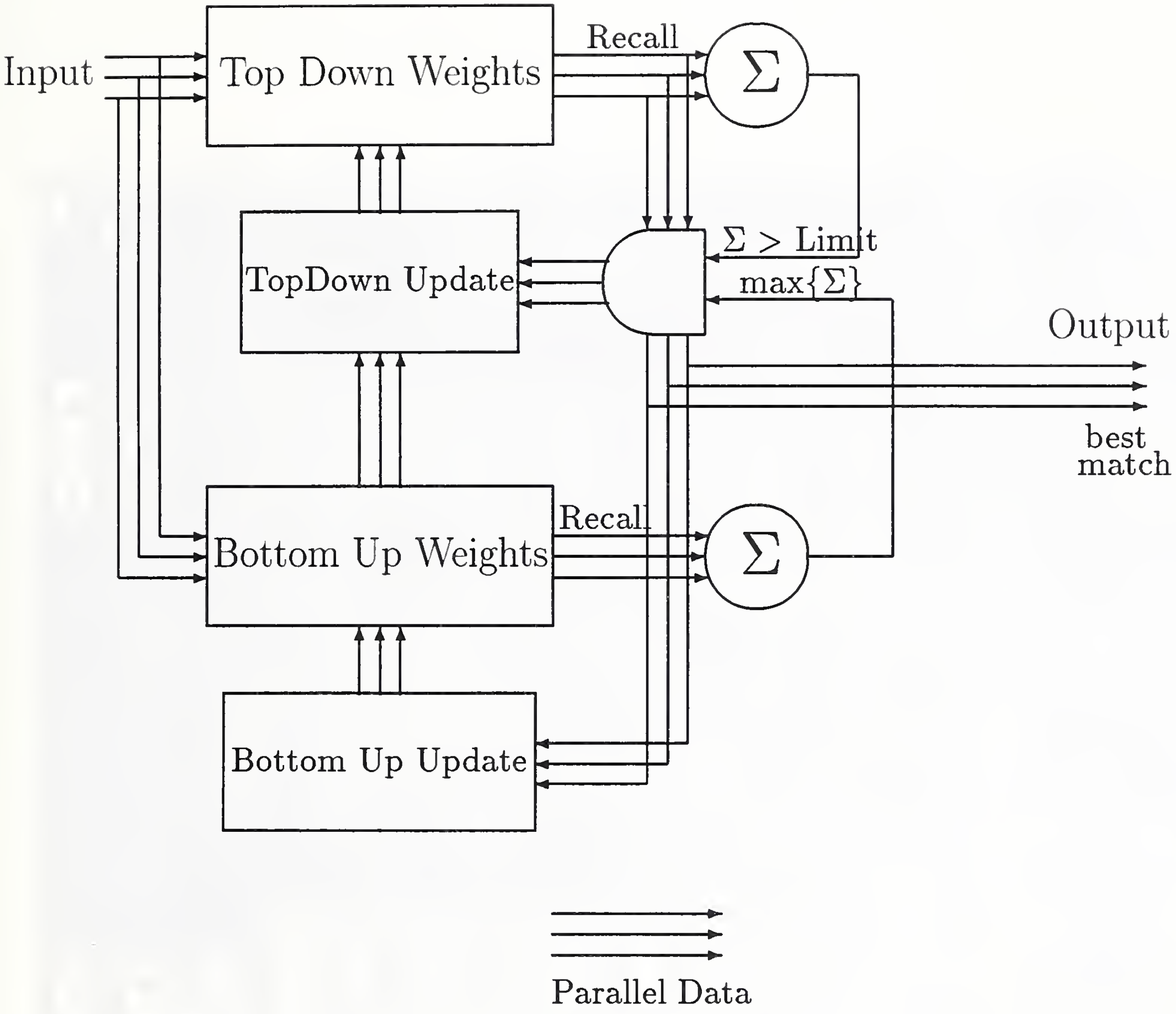

Figure 1: ART-1 architecture as implemented on the parallel processor array. The upper summation over the top-down weights is carried out using equation 3 to produce $\bar{T}$. The lower summation over the bottom up weights is carried out using equation 1 to produce μ. The Limit shown on the two gates is the learning threshold, ρ.

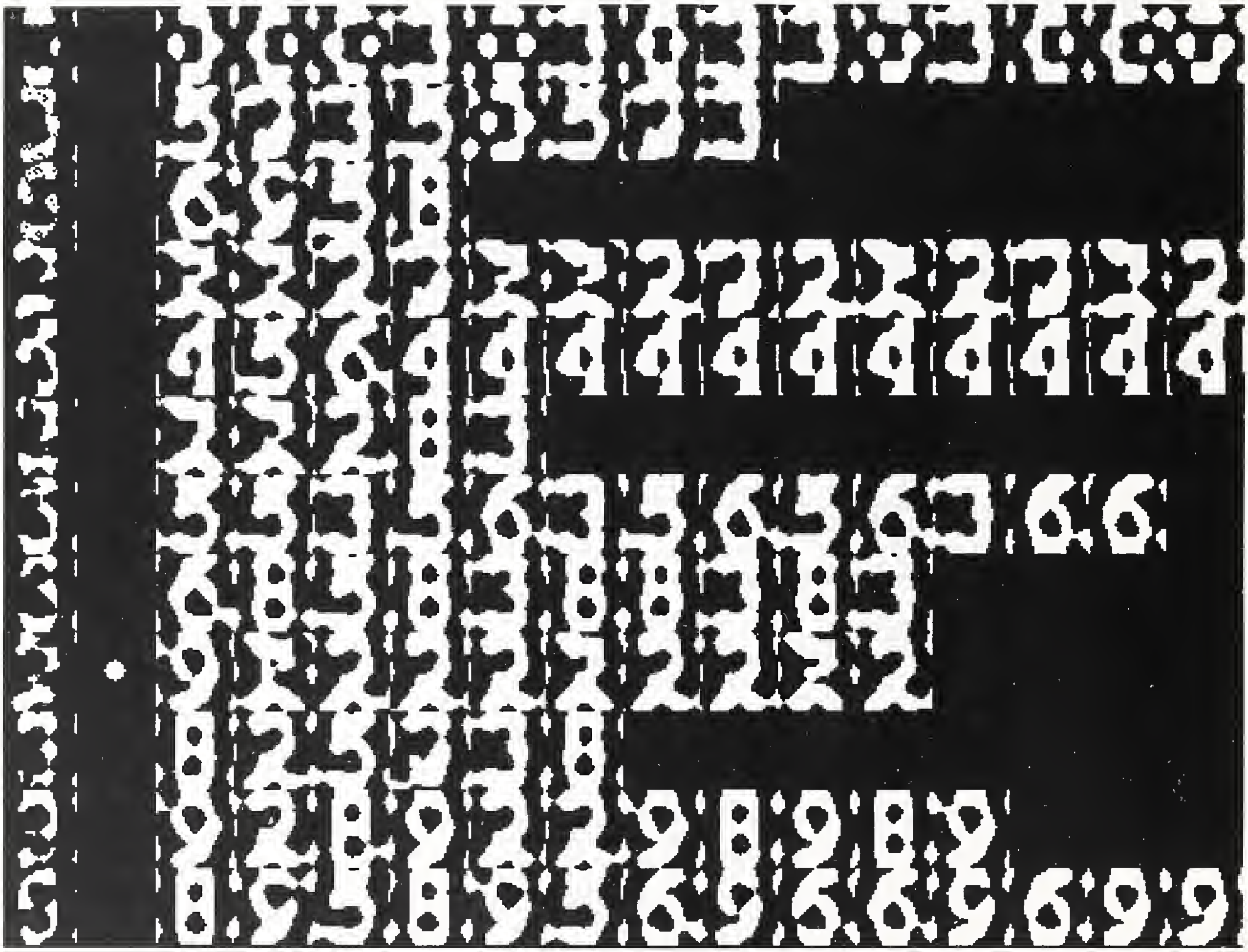

Figure 2: ART-1 feature extraction for machine print with no filter and with Boltzmann optimization for a $\bar{T}$, = 0.7.

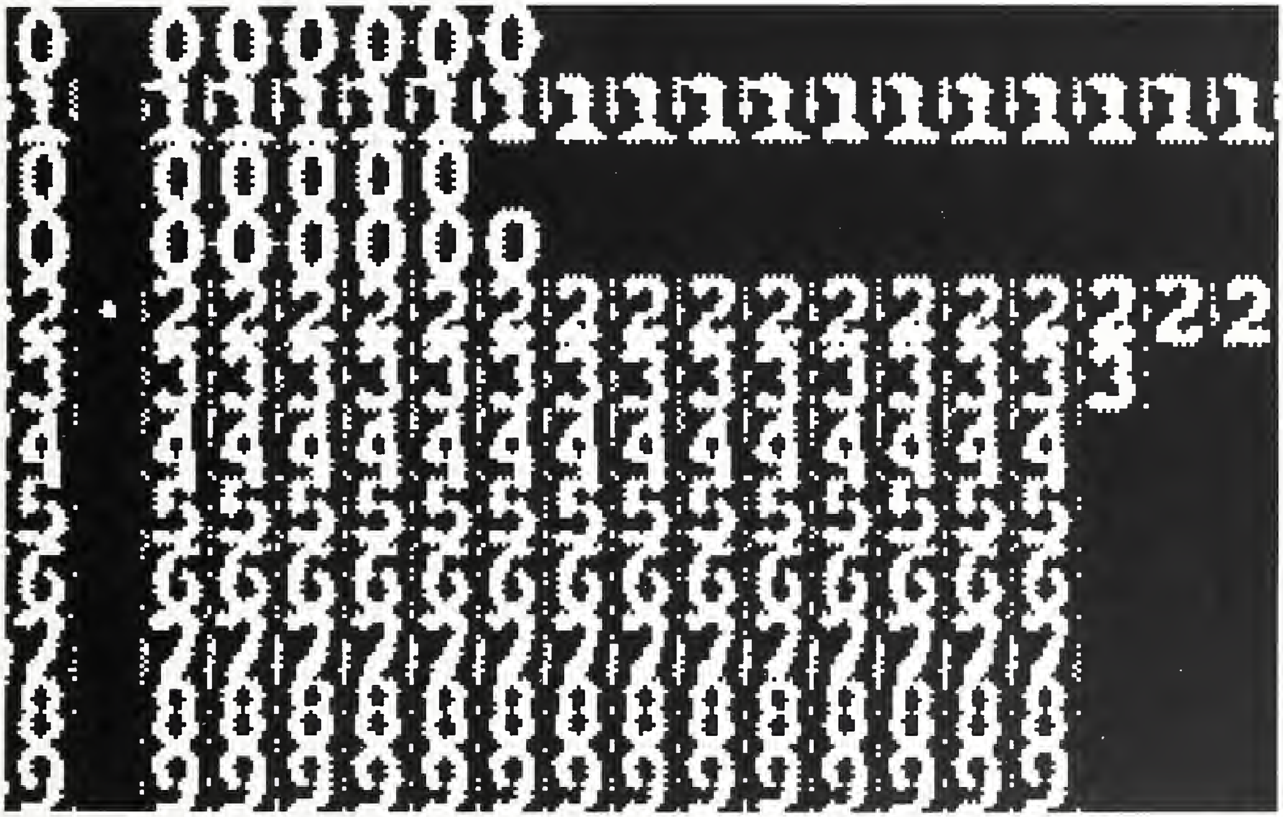

Figure 3: ART-1 feature extraction for machine print with no filter and without Boltzmann optimization.

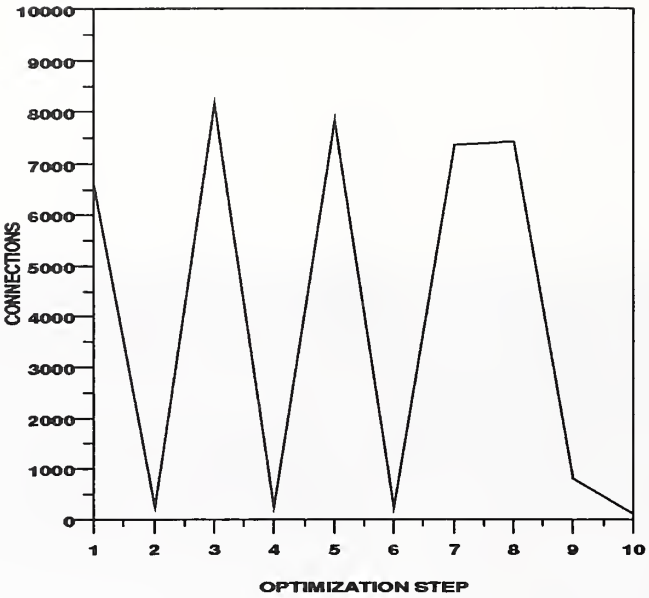

Figure 4: Changes in number of connections during the optimization process $T = 0.5$ and $\rho = 0.7$

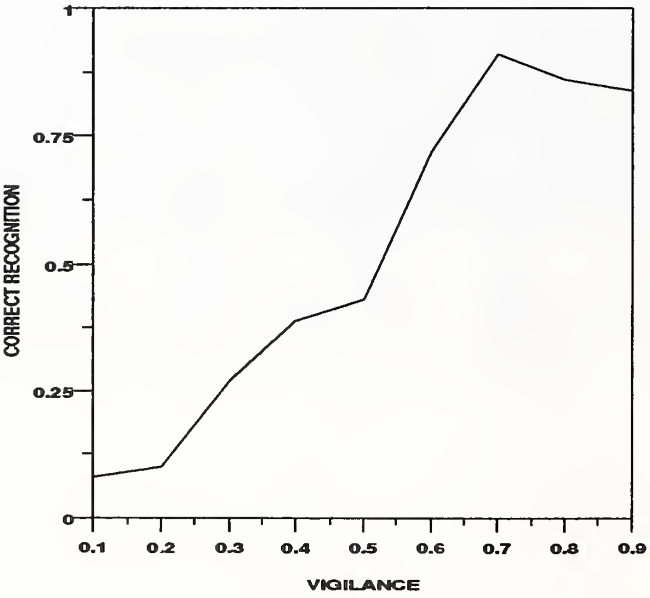

Figure 5: Network testing accuracy at $T = 0.5$

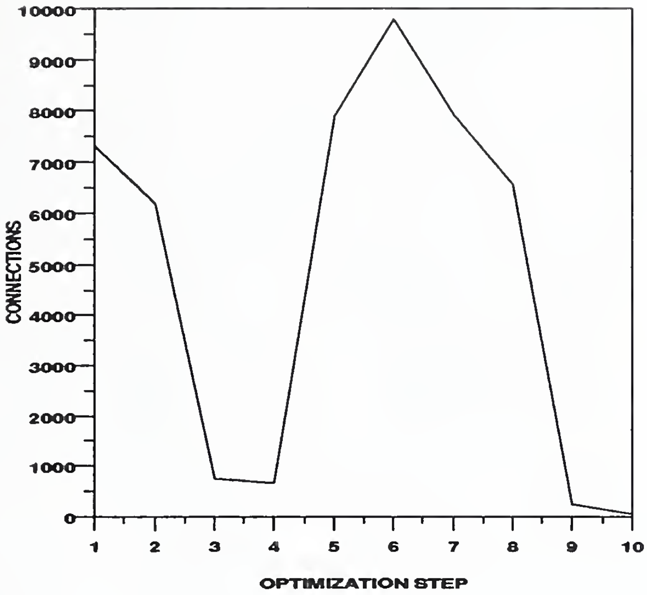

Figure 6: Changes in number of connections during the optimization process $T = 0.8$ and $\rho = 0.8$

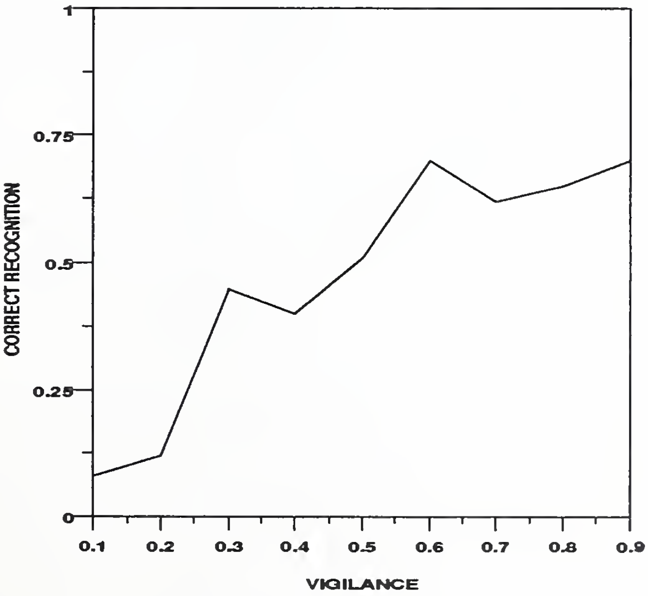

Figure 7: Network testing accuracy at $T = 0.8$